SRI International

LOCALIZING EXPRESSION OF AMBIGUITY

Technical Note 428

November 30, 1987

By: John Bear, Computer Scientist
              and
       Jerry R. Hobbs, Sr. Computer Scientist

       Artificial Intelligence Center
       Computer and Information Sciences Division

**APPROVED FOR PUBLIC RELEASE:
DISTRIBUTION UNLIMITED**

		Form Approved
Report Documentation Page		*OMB No. 0704-0188*

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **30 NOV 1987**	2. REPORT TYPE	3. DATES COVERED **00-11-1987 to 00-11-1987**
4. TITLE AND SUBTITLE **Localizing Expression of Ambiguity**		5a. CONTRACT NUMBER
		5b. GRANT NUMBER
		5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)		5d. PROJECT NUMBER
		5e. TASK NUMBER
		5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **SRI International,333 Ravenswood Avenue,Menlo Park,CA,94025**		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S)
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES **20**	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**			

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

Localizing Expression of Ambiguity

John Bear and Jerry R. Hobbs
Artificial Intelligence Center
SRI International

Abstract

In this paper we describe an implemented program for localizing the expression of many types of syntactic ambiguity, in the logical forms of sentences, in a manner convenient for subsequent inferential processing. Among the types of ambiguities handled are prepositional phrases, very compound nominals, adverbials, relative clauses, and preposed prepositional phrases. The algorithm we use is presented, and several possible shortcomings and extensions of our method are discussed.

1   Introduction

Ambiguity is a problem in any natural language processing system. Large grammars tend to produce large numbers of alternative analyses for even relatively simple sentences. Furthermore, as is well known, syntactic information may be insufficient for selecting a best reading. It may take semantic knowledge of arbitrary complexity to decide which alternative to choose.

In the TACITUS project [Hobbs, 1986; Hobbs and Martin, 1987] we are developing a pragmatics component which, given the logical form of a sentence, uses world knowledge to solve various interpretation problems, the resolution of syntactic ambiguity among them. Sentences are translated into logical form by the DIALOGIC system for syntactic and semantic analysis [Grosz et al., 1982]. In this paper we describe how information about alternative parses is passed concisely from DIALOGIC to the pragmatics component, and more generally, we discuss a method of localizing the representation of syntactic ambiguity in the logical form of a sentence.

One possible approach to the ambiguity problem would be to produce a set of logical forms for a sentence, one for each parse tree, and to send them one at a time to the pragmatics component. This involves considerable

1

duplication of effort if the logical forms are largely the same and differ only with respect to attachment. A more efficient approach is to try to localize the information about the alternate possibilities.

Instead of feeding two logical forms, which differ only with respect to an attachment site, to a pragmatics component, it is worthwhile trying to condense the information of the two logical forms together into one expression with a disjunction inside it representing the attachment ambiguity. That one expression may then be given to a pragmatics component with the effect that parts of the sentence that would have been processed twice are now processed only once. The savings can be considerably more dramatic when a set of five or ten or twenty logical forms can be reduced to one, as is often the case.

In effect, this approach translates the syntactic ambiguity problem into a highly constrained coreference problem. It is as though we translated the sentence in (1) into the two sentences in (2)

(1)   John drove down the street in a car.
(2)   John drove down the street. It was in a car.

where we knew "it" had to refer either to the street or to the driving. Since coreference is one of the phenomena the pragmatics component is designed to cope with [Hobbs and Martin, 1987], such a translation represents progress toward a solution.

The rest of this paper describes the procedures we use to produce a reduced set of logical forms from a larger set. The basic strategy hinges on the idea of a neutral representation [Hobbs, 1982]. This is similar to the idea behind Church's Pseudo-attachment [Church, 1980], Pereira's Rightmost Normal Form [Pereira, 1983], and what Rich et al. refer to as the Procrastination Approach to parsing [Rich, Barnett, Wittenburg, and Whittemore, 1986]. However, by expressing the ambiguity as a disjunction in logical form, we put it into the form most convenient for subsequent inferential processing.

2   Range of Phenomena

2.1   Attachment Possibilities

There are three representative classes of attachment ambiguities, and we have implemented our approach to each of these. For each class, we give

2

representative examples and show the relevant logical form fragments that encode the set of possible attachments.

In the first class are those constituents that may attach to either nouns or verbs.

(3) John saw the man with the telescope.

The prepositional phrase (PP) "with the telescope" can be attached either to "the man" or to "saw". If m stands for the man, t for the telescope, and e for the seeing event, the neutral logical form for the sentence includes

$$\ldots \wedge with(y,t) \wedge [y = m \vee y = e] \wedge \ldots$$

That is, something y is with the telescope, and it is either the man or the seeing event.

Gerund modifiers may also modify nouns and verbs, resulting in ambiguities like that in the sentence

I saw the Grand Canyon, flying to New York.

Their treatment is identical to that of PPs. If g is the Grand Canyon, n is New York, and e is the seeing event, the neutral logical form will include

$$\ldots \wedge fly(y,n) \wedge [y = g \vee y = e] \wedge \ldots$$

That is, something y is flying to New York, and it is either the Grand Canyon or the seeing event.[1]

In the second class are those constituents that can only attach to verbs, such as adverbials.

George said Sam left his wife yesterday.

Here "yesterday" can modify the saying or the leaving but not "his wife". Suppose we take *yesterday* to be a predicate that applies to events and specifies something about their times of occurrence, and suppose e_1 is the leaving event and e_2 the saying event. Then the neutral logical form will include

$$\ldots \wedge yesterday(y) \wedge [y = e_1 \vee y = e_2] \wedge \ldots$$

---

[1] If the seeing event is flying to New York we can infer that the seer is also flying to New York.

That is, something y was yesterday and it is either the leaving event or the saying event.

Related to this is the case of a relative clause where the preposed constituent is a PP, which could have been extracted from any of several embedded clauses. In

> That was the week during which George thought Sam told his
> wife he was leaving,

the thinking, the telling, or the leaving could have been during the week. Let w be the week, e_1 the thinking, e_2 the telling, and e_3 the leaving. Then the neutral logical form will include

$$\ldots \wedge \; during(y, w) \wedge [y = e_1 \vee y = e_2$$
$$\vee \; y = e_3] \wedge \ldots$$

That is, something y was during the week, and y is either the thinking, the telling, or the leaving.

The third class contains those constituents that may only attach to nouns, e.g., relative clauses.

> This component recycles the oil that flows through the compres-
> sor that is still good.

The second relative clause, "that is still good," can attach to "compressor", or "oil", but not to "flows" or "recycles". Let o be the oil and c the compressor. Then, ignoring "still", the neutral logical form will include

$$\ldots \wedge \; good(y) \wedge [y = c \vee y = o] \wedge \ldots$$

That is, something y is still good, and y is either the compressor or the oil.

Similar to this are the compound nominal ambiguities, as in

> He inspected the oil filter element.

"Oil" could modify either "filter" or "element". Let o be the oil, f the filter, e the element, and nn the implicit relation that is encoded by the nominal compound construction. Then the neutral logical form will include

$$\ldots \wedge \; nn(f, e) \wedge nn(o, y) \wedge [y = f \vee y = e] \wedge \ldots$$

4

That is, there is some implicit relation nn between the filter and the element, and there is another implicit relation nn between the oil and something y, where y is either the filter or the element.

Our treatment of all of these types of ambiguity has been implemented.

In fact, the distinction we base the attachment possibilities on is not that between nouns and verbs, but that between event variables and entity variables in the logical form. This means that we would generate logical forms encoding the attachment of adverbials to event nominalizations in those cases where the event nouns are translated with event variables. Thus in

I read about Judith's promotion last year.

"last year" would be taken as modifying either the promotion or the reading, if "promotion" were represented by an event variable in the logical form.

2.2   Single or Multiple Parse Trees

In addition to classifying attachment phenomena in terms of which kind of constituent something may attach to, there is another dimension along which we need to classify the phenomena: does the DIALOGIC parser produce all possible parses, or only one? For some regular structural ambiguities, such as very compound nominals, and the "during which" examples, only a single parse is produced. In this case it is straightforward to produce from the parse a neutral representation encoding all the possibilities. In the other cases, however, such as (nonpreposed) PPs, adverbials, and relative clauses, DIALOGIC produces an exhaustive (and sometimes exhausting) list of the different possible structures. This distinction is an artifact of our working in the DIALOGIC system. It would be preferable if there were only one tree constructed which was somehow neutral with respect to attachment. However, the DIALOGIC grammar is large and complex, and it would have been difficult to implement such an approach. Thus, in these cases, one of the parses, the one corresponding to right association [Kimball, 1973], is selected, and the neutral representation is generated from that. This makes it necessary to suppress redundant readings, as described below. (In fact, limited heuristics for suppressing multiple parse trees have recently been implemented in DIALOGIC.)

5

2.3 Thematic Role Ambiguities

Neutral representations are constructed for one other kind of ambiguity in the TACITUS system—ambiguities in the thematic role or case of the arguments. In the sentence

It broke the window.

we don't know whether "it" is the agent or the instrument. Suppose the predicate *break* takes three arguments, an agent, a patient, and an instrument, and suppose x is whatever is referred to by "it" and w is the window. Then the neutral logical form will include

$$\ldots \wedge break(y_1, w, y_2) \wedge [y_1 = x \vee y_2 = x] \wedge \ldots$$

That is, something y_1 breaks the window with something else y_2, and either y_1 or y_2 is whatever is referred to by "it".[2]

2.4 Ambiguities Not Handled

There are other types of structural ambiguity about which we have little to say. In

They will win one day in Hawaii,

one of the obvious readings is that "one day in Hawaii" is an adverbial phrase. However, another perfectly reasonable reading is that "one day in Hawaii" is the direct object of the verb "win". This is due to the verb having more than one subcategorization frame that could be filled by the surrounding constituents. It is the existence of this kind of ambiguity that led to the approach of not having DIALOGIC try to build a single neutral representation in all cases. A neutral representation for such sentences, though possible, would be very complicated.

Similarly, we do not attempt to produce neutral representations for fortuitous or unsystematic ambiguities such as those exhibited in sentences like

They are flying planes.

Time flies like an arrow.

Becky saw her duck.

---

[2]The treatment of thematic role ambiguities has been implemented by Paul Martin as part of the interface between DIALOGIC and the pragmatic processes of TACITUS that translates the logical forms of the sentences into a canonical representation.

6

2.5 Resolving Ambiguities

It is beyond the scope of this paper to describe the pragmatics processing that is intended to resolve the ambiguities (see Hobbs and Martin, 1987). Nevertheless, we discuss one nontrivial example, just to give the reader a feel for the kind of processing it is. Consider the sentence

> We retained the filter element for future analysis.

We would like the system to infer that the right reading is that "for future analysis" modifies the verb "retain" and not the NP "filter element".

Let r be the retaining event, f the filter element, and a the analysis. Then the logical form for the sentence will include

$$\ldots \wedge\ for(y, a) \wedge [y = f \vee y = r] \wedge \ldots$$

The predicate *for*, let us say, requires the relation $enable(y, a)$ to obtain between its arguments. That is, if y is for a, then either y or something coercible from y must somehow enable a or something coercible from a. The TACITUS knowledge base contains axioms encoding the fact that having something is a prerequisite for analyzing it and the fact that a retaining is a having. y can thus be equal to r, which is consistent with the constraints on y. On the other hand, any inference that the filter element enables the analysis will be much less direct, and consequently will not be chosen.

3 The Algorithm

3.1 Finding Attachment Sites

The logical forms (LFs) that are produced from each of the parse trees are given to an attachment-finding program which adds, or makes explicit, information about possible attachment sites. Where this makes some LFs redundant, as in the prepositional phrase case, the redundant LFs are then eliminated.

For instance, for the sentence in (4),

(4)   John saw the man in the park with the telescope.

DIALOGIC produces five parse trees, and five corresponding logical forms. When the attachment-finding routine is run on an LF, it annotates the LF with information about a set of variables that might be the subject (i.e., the attachment site) of each PP.

The example below shows the LFs for one of the five readings before
and after the attachment-finding routine is run on it. They are somewhat
simplified for the purposes of exposition. In this notation, a proposition
is a predicate followed by one or more arguments. An argument is a vari-
able or a complex term. A complex term is a variable followed by a "such
that" symbol "$|$", followed by a conjunction of one or more propositions.[3]
Complex terms are enclosed in square brackets for readability. Events are
represented by event variables, as in [Hobbs, 1985], so that $see'(e_1, x_1, x_2)$
means e_1 is a seeing event by x_1 of x_2.

One of sentence (4)'s LFs before attachment-finding is

$$past([e_1 \mid see'(e_1,$$
$$[x_1 \mid John(x_1)],$$
$$[x_2 \mid man(x_2) \wedge$$
$$in(x_2, [x_3 \mid park(x_3) \wedge$$
$$with(x_3, [x_4 \mid telescope(x_4)])])])]))$$

The same LF after attachment-finding is

$$past([e_1 \mid see'(e_1,$$
$$[x_1 \mid John(x_1)],$$
$$[x_2 \mid man(x_2) \wedge$$
$$in([y_1 \mid y_1 = x_2 \vee y_1 = e_1],$$
$$[x_3 \mid park(x_3) \wedge$$
$$with([y_2 \mid y_2 = x_3 \vee y_2 = x_2 \vee y_2 = e_1],$$
$$[x_4 \mid telescope(x_4)])])])]))$$

A paraphrase of the latter LF in English would be something like this:
There is an event e_1 that happened in the past; it is a seeing event by x_1
who is John, of x_2 who is the man; something y_1 is in the park, and that
something is either the man or the seeing event; something y_2 is with a
telescope, and that something is the park, the man, or the seeing event.

The procedure for finding possible attachment sites in order to modify
a logical form is as follows. The program recursively descends an LF, and
keeps lists of the event and entity variables that initiate complex terms.
Event variables associated with tenses are omitted. When the program
arrives at some part of the LF that can have multiple attachment sites,

---

[3]This notation can be translated into a Russellian notation, with the consequent loss
of information about grammatical subordination, by repeated application of the transfor-
mation $p(x \mid Q) \Rightarrow p(x) \wedge Q$.

it replaces the explicit argument by an existentially quantified variable y, determines whether it can be an event variable, an entity variable, or either, and then encodes the list of possibilities for what y could equal.

3.2 Eliminating Redundant Logical Forms

In those cases where more than one parse tree, and hence more than one logical form, is produced by DIALOGIC, it is necessary to eliminate redundant readings. In order to do this, once the attachment possibilities are registered, the LFs are flattened (thus losing temporarily the grammatical subordination information), and some simplifying preprocessing is done. Each of the flattened LFs is compared with the others. Any LF that is subsumed by another is discarded as redundant. One LF subsumes another if the two LFs are the same except that the first has a list of possible attachment sites that includes the corresponding list in the second. For example, one LF for sentence (3) says that "with the telescope" can modify either "saw" or "the man", and one says that it modifies "saw". The first LF subsumes the second, and the second is discarded and not compared with any other LFs. Thus, although the LFs are compared pairwise, if all of the ambiguity is due to only one attachment indeterminacy, each LF is looked at only once.

Frequently, only some of the alternatives may be thrown out. For

Andy said he lost yesterday

after attachment-finding, one logical form allows "yesterday" to be attached to either the saying or the losing, while another attaches it only to the saying. The second is subsumed by the first, and thus discarded. However, there is a third reading in which "yesterday" is the direct object of "lost" and this neither subsumes nor is subsumed by the others and is retained.

4 Lost Information

4.1 Crossing Dependencies

Our attachment-finding routine constructs a logical form that describes all of the standard readings of a sentence, but it also describes some nonstandard readings, namely those corresponding to parse trees with crossing branches, or crossing dependencies. An example would be a reading of (4) in which the seeing was in the park and the man was with the telescope.

For small numbers of possible attachment sites, this is an acceptable result. If a sentence is two-ways ambiguous (due just to attachment), we get no wrong readings. If it is five-ways ambiguous on the standard analysis, we get six readings. However, in a sentence with a sequence of four PPs, the standard analysis (and the DIALOGIC parser) get 42 readings, whereas our single disjunctive LF stands for 120 different readings.

Two things can be said about what to do in these cases where the two approaches diverge widely. We could argue that sentences with such crossing dependencies do exist in English. There are some plausible sounding examples.

> Specify the length, in bytes, of the word.

> Kate saw a man on Sunday with a wooden leg.

In the first, the phrase "in bytes" modifies "specify", and "of the word" modifies "the length". In the second, "on Sunday" modifies "saw" and "with a wooden leg" modifies "a man". Stucky [1987] argues that such examples are acceptable and quite frequent.

On the other hand, if one feels that these putative examples of crossing dependencies can be explained away and should be ruled out, there is a way to do it within our framework. One can encode in the LFs a crossing-dependencies constraint, and consult that constraint when doing the pragmatic processing.

To handle the crossing-dependencies constraint (which we have not yet implemented), the program would need to keep the list of the logical variables it constructs. This list would contain three kinds of variables, event variables, entity variables, and the special variables (the y's in the LFs above) representing attachment ambiguities. The list would keep track of the order in which variables were encountered in descending the LF. A separate list of just the special y variables also needs to be kept. The strategy would be that in trying to resolve referents, whenever one tries to instantiate a y variable to something, the other y variables need to be checked, in accordance with the following constraint:

> There cannot be y_1, y_2 in the list of y's such that $B(y_1) <$ $B(y_2) < y_1 < y_2$, where $B(y_i)$ is the proposed variable to which y_i will be bound or with which it will be coreferential, and the $<$ operator means "precedes in the list of variables".

This constraint handles a single phrase that has attachment ambiguities.

10

It also works in the case where there is a string of PPs in the subject NP, and then a string of PPs in the object NP, as in

> The man with the telescope in the park lounged on the bank of
> a river in the sun.

With the appropriate crossing-dependency constraints, the logical form for this would be[4]

$past([e_1 \mid lounge'(e_1,$
$\qquad\qquad [x_1 \mid man(x_1) \wedge$
$\qquad\qquad\qquad with([y_1 \mid y_1 = x_1 \vee y_1 = e_1],$
$\qquad\qquad\qquad\qquad [x_2 \mid telescope(x_2) \wedge$
$\qquad\qquad\qquad\qquad\qquad in([y_2 \mid y_2 = x_2 \vee y_2 = x_1 \vee y_2 = e_1],$
$\qquad\qquad\qquad\qquad\qquad\qquad [x_3 \mid park(x_3)])])]) \wedge$
$\qquad\qquad on(e_1,$
$\qquad\qquad\quad [x_4 \mid bank(x_4)$
$\qquad\qquad\qquad\quad of([y_3 \mid y_3 = x_4 \vee y_3 = e_1],$
$\qquad\qquad\qquad\qquad [x_5 \mid river(x_5) \wedge$
$\qquad\qquad\qquad\qquad\qquad in([y_4 \mid y_4 = x_5 \vee y_4 = x_4 \vee y_4 = e_1],$
$\qquad\qquad\qquad\qquad\qquad\qquad [x_6 \mid sun(x_6)])])]) \wedge$
$\qquad\qquad crossing\text{-}info(< e_1, x_1, y_1, x_2, y_2, x_3 >, \{y_1, y_2\}) \wedge$
$\qquad\qquad crossing\text{-}info(< e_1, x_4, y_3, x_5, y_4, x_6 >, \{y_3, y_4\})])$

4.2   Noncoreference Constraints

One kind of information that is provided by the DIALOGIC system is information about coreference and noncoreference insofar as it can be determined from syntactic structure. Thus, the logical form for

> John saw him.

includes the information that "John" and "him" cannot be coreferential. This interacts with our localization of attachment ambiguity. Consider the sentence,

> John returned Bill's gift to him.

---

[4]We are assuming "with the telescope" and "in the park" can modify the lounging, which they certainly can if we place commas before and after them.

11

If we attach "to him" to "gift", "him" can be coreferential with "John" but it cannot be coreferential with "Bill". If we attach it to "returned", "him" can be coreferential with "Bill" but not with "John". It is therefore not enough to say that the "subject" of "to" is either the gift or the returning. Each alternative carries its own noncoreference constraints with it. We do not have an elegant solution to this problem. We mention it because, to our knowledge, this interaction of noncoreference constraints and PP attachment has not been noticed by other researchers taking similar approaches.

5   A Note on Literal Meaning

There is an objection one could make to our whole approach. If our logical forms are taken to be a representation of the "literal meaning" of the sentence, then we would seem to be making the claim that the literal meaning of sentence (2) is "Using a telescope, John saw a man, or John saw a man who had a telescope," whereas the real situation is that either the literal meaning is "Using a telescope, John saw a man," or the literal meaning is "John saw a man who had a telescope." The disjunction occurs in the metalanguage, whereas we may seem to be claiming it is in the language.

The misunderstanding behind this objection is that the logical form is not intended to represent "literal meaning". There is no general agreement on precisely what constitutes "literal meaning", or even whether it is a coherent notion. In any case, few would argue that *the* meaning of a sentence could be determined on the basis of syntactic information alone. The logical forms produced by the DIALOGIC system are simply intended to encode all of the information that syntactic processing can extract about the sentence. Sometimes the best we can come up with in this phase of the processing is disjunctive information about attachment sites, and that is what the LF records.

6   Future Extensions

6.1   Extending the Range of Phenomena

The work that has been done demonstrates the feasibility of localizing in logical form information about attachment ambiguities. There is some mundane programming to do to handle the cases similar to those described here,

e.g., other forms of postnominal modification. There is also the crossing-dependency constraint to implement.

The principal area in which we intend to extend our approach is various kinds of conjunction ambiguities. Our approach to some of these cases is quite similar to what we have presented already. In the sentence,

(5)   Mary told us John was offended and George left the
        party early.

it is possible for George's leaving to be conjoined with either John's being offended or Mary's telling. Following Hobbs [1985], conjunction is represented in logical form by the predicate and' taking a self argument and two event variables as its arguments. In (5) suppose e_1 stands for the telling, e_2 for the being offended, e_3 for the leaving, and e_0 for the conjunction. Then the neutral representation for (5) would include

$$and'(e_0, y_0, e_3) \wedge tell'(e_1, M, y_1)$$
$$\wedge ((y_0 = e_1 \wedge y_1 = e_2) \vee (y_0 = e_2 \wedge y_1 = e_0))$$

That is, there is a conjunction e_0 of y_0 and the leaving e_3; there is a telling e_1 by Mary of y_1; and either y_0 is the telling e_1 and y_1 is the being offended e_2, or y_0 is the being offended e_2 and y_1 is the conjunction e_0.

A different kind of ambiguity occurs in noun phrase conjunction. In

(6) Where are the British and American ships?

there is a set of British ships and a disjoint set of American ships, whereas in

(7) Where are the tall and handsome men?

the natural interpretation is that a single set of men is desired, consisting of men who are both tall and handsome.

In TACITUS, noun phrase conjunction is encoded with the predicate $andn$, taking three sets as its arguments. The expression $andn(s_1, s_2, s_3)$ means that the set s_1 is the union of sets s_2 and s_3.[5] Following Hobbs [1983], the representation of plurals involves a set and a typical element of the set, or a reified universally quantified variable ranging over the elements of the set. Properties like cardinality are properties of the set itself, while properties

---

[5]If either s_1 or s_2 is not a set, the singleton set consisting of just that element is used instead.

that hold for each of the elements are properties of the typical element. An axiom schema specifies that any properties of the typical element are inherited by the individual, actual elements.[6] Thus, the phrase "British and American ships" is translated into the set s_1 such that

$$andn(s_1, s_2, s_3) \land typelt(x_1, s_1) \land ship(x_1)$$
$$\land typelt(x_2, s_2) \land British(x_2)$$
$$\land typelt(x_3, s_3) \land American(x_3)$$

That is, the typical element x_1 of the set s_1 is a ship, and s_1 is the union of the sets s_2 and s_3, where the typical element x_2 of s_2 is British, and the typical element x_3 of s_3 is American.

The phrase "tall and handsome men" can be represented in the same way.

$$andn(s_1, s_2, s_3) \land typelt(x_1, s_1) \land man(x_1)$$
$$\land typelt(x_2, s_2) \land tall(x_2)$$
$$\land typelt(x_3, s_3) \land handsome(x_3)$$

Then it is a matter for pragmatic processing to discover that the set s_2 of tall men and the set s_3 of handsome men are in fact identical.

In this representational framework, the treatment given to the kind of ambiguity illustrated in

I like intelligent men and women.

resembles the treatment given to attachment ambiguities. The neutral logical form would include

$$\ldots \land andn(s_1, s_2, s_3) \land typelt(x_1, s_1)$$
$$\land typelt(x_2, s_2) \land man(x_2)$$
$$\land typelt(x_3, s_3) \land woman(x_3)$$
$$\land intelligent(y) \land [y = x_1 \lor y = x_2]$$

That is, there is a set s_1, with typical element x_1, which is the union of sets s_2 and s_3, where the typical element x_2 of s_2 is a man and the typical element x_3 of s_3 is a woman, and something y is intelligent, where y is either the typical element x_1 of s_1 (the typical person) or the typical element x_2 of s_2 (the typical man).

Ambiguities in conjoined compound nominals can be represented similarly. The representation for

---

[6]The reader may with some justification feel that the term *"typical* element" is ill-chosen. He or she is invited to suggest a better term.

oil pump and filter

would include

$$\ldots \wedge\ andn(s, p, f)\ \wedge\ typelt(x, s)\ \wedge\ pump(p)$$
$$\wedge\ filter(f)\ \wedge\ oil(o)\ \wedge\ nn(o, y)$$
$$\wedge\ [y = p\ \vee\ y = x]$$

That is, there is a set s, with typical element x, composed of the elements p and f, where p is a pump and f is a filter, and there is some implicit relation nn between some oil o and y, where y is either the pump p or the typical element x or s. (In the latter case, the axiom in the TACITUS system's knowledge base,

$$(\forall\ w, x, y, z, s)nn(w, x)\ \wedge\ typelt(x, s)$$
$$\wedge\ andn(s, y, z)$$
$$\equiv\ nn(w, y)\ \wedge\ nn(w, z)$$

allows the nn relation to be distributed to the two conjuncts.)

6.2 Ordering Heuristics

So far we have only been concerned with specifying the set of possible attachment sites. However, it is true, empirically, that certain attachment sites can be favored over others, strictly on the basis of syntactic (and simple semantic) information alone.[7]

For example, for the prepositional phrase attachment problem, an informal study of several hundred examples suggests that a very good heuristic is obtained by using the following three principles: (1) favor right association; (2) override right association if (a) the PP is temporal and the second nearest attachment site is a verb or event nominalization, or (b) if the preposition typically signals an argument of the second nearest attachment site (verb or relational noun) and not of the nearest attachment site; (3) override right association if a comma (or comma intonation) separates the PP from the nearest attachment site. The preposition"of" should be treated specially; for "of" PPs, right association is correct over 98% of the time.

There are two roles such a heuristic ordering of possibilities can play. In a system without sophisticated semantic or pragmatic processing, the favored attachment could simply be selected. On the other hand, in a system such

---

[7]There is a vast literature on this topic. For a good introduction, see Dowty, Karttunen, and Zwicky [1985].

as TACITUS in which complex inference procedures access world knowledge in interpreting a text, the heuristic ordering can influence an allocation of computational resources to the various possibilities.

Acknowledgements

References

[1] Dowty, David, Lauri Karttunen, and Arnold Zwicky (1985) *Natural Language Parsing*, Cambridge University Press.

[2] Church, Kenneth (1980) "On Memory Limitations in Natural Language Processing", Technical Note, MIT Computer Science Lab, MIT.

[3] Church, Kenneth, and Ramesh Patil (1982) "Coping with Syntactic Ambiguity or How to Put the Block in the Box on the Table", *AJCL*, Vol 8, No 3-4.

[4] Grosz, Barbara, Norman Haas, Gary Hendrix, Jerry Hobbs, Paul Martin, Robert Moore, Jane Robinson, Stanley Rosenschein (1982) "DIALOGIC: A Core Natural-Language Processing System", Technical Note 270, Artificial Intelligence Center, SRI International.

[5] Hirst, Graeme (1986) "Semantic Interpretation and Ambiguity", to appear in *Artificial Intelligence*.

[6] Hobbs, Jerry (1982) "Representing Ambiguity", *Proceedings of the First West Coast Conference on Formal Linguistics*, Stanford University Linguistics Department, pp. 15-28.

[7] Hobbs, Jerry (1983) "An Improper Approach to Quantification in Ordinary English", *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, Cambridge, Massachusetts, pp. 57-63.

[8] Hobbs, Jerry (1985) "Ontological Promiscuity", *Proceedings of the 23rd Annual Meeting of the Association for Computational Linguistics*, Chicago, Illinois, pp. 61-69.

[9] Hobbs, Jerry (1986) "Overview of the TACITUS Project", *CL*, Vol. 12, No. 3.

[10] Hobbs, Jerry, and Paul Martin (1987) "Local Pragmatics", *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, Milano, Italy, pp. 520-523.

[11] Kimball, John (1973) "Seven Principles of Surface Structure Parsing", *Cognition*, Vol. 2, No. 1, pp. 15-47.

[12] Pereira, Fernando (1983) "Logic for Natural Language Analysis", Technical Note 275, Artificial Intelligence Center, SRI International.

[13] Rich, Elaine, Jim Barnett, Kent Wittenburg, and Greg Whittemore (1986) "Ambiguity and Procrastination in NL Interfaces", Technical Note HI-073-86, MCC.

[14] Stucky, Susan (1987) "Configurational Variation in English: A Study of Extraposition and Related Matters", in *Syntax and Semantics: Discontinuous Constituency*, Vol. 20, edited by G. Huck and A. Ojeda, Academic Press.

Appendix

John saw the man with the telescope.

Logical Form before Attachment-Finding:

```
((PAST
  (SELF E11)
  (SUBJECT
   (E3
    (SEE
     (SELF E3)
     (SUBJECT (X1 (JOHN (SELF E2) (SUBJECT X1))))
     (OBJECT (X4 (MAN (SELF E5) (SUBJECT X4))
                 (WITH (SELF E6)
                       ; Here [with] modifies [man]
                       (PP-SUBJECT X4)
                       (OBJECT (X7 (TELESCOPE (SELF E8)
                                              (SUBJECT X7))
                                   (THE (SELF E9)
                                        (SUBJECT X7))
                                   (NOT= (NP X7)
                                         (ANTES (X4))))))
                 (THE (SELF E10) (SUBJECT X4))
                 (NOT= (NP X4) (ANTES (X1)))))))))))
```

Logical Form after Attachment-Finding:

```
((PAST
  (SELF E11)
  (SUBJECT
   (E3
    (SEE
     (SELF E3)
     (SUBJECT (X1 (JOHN (SELF E2) (SUBJECT X1))))
     (OBJECT (X4 (MAN (SELF E5) (SUBJECT X4))
                 (WITH (SELF E6)
                       ; Here [with] modifies [man] or [saw]
                       (SUBJECT (Y14 (?= (NP Y14)
                                         (ANTES (X4 E3)))))
                       (OBJECT (X7 (TELESCOPE (SELF E8)
                                              (SUBJECT X7))
                                   (THE (SELF E9)
                                        (SUBJECT X7))
                                   (NOT= (NP X7)
                                         (ANTES (X4))))))
                 (THE (SELF E10) (SUBJECT X4))
                 (NOT= (NP X4) (ANTES (X1)))))))))))
```